UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P O Box 1450
Alexandria, Virginia 22313-1450
www uspto gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/881,628	06/13/2001	Enrique Musoll	MIPS.0174-00-US	8580

23669 7590 05/02/2008
HUFFMAN LAW GROUP, P.C.
1900 MESA AVE.
COLORADO SPRINGS, CO 80906

EXAMINER	
MEW, KEVIN D	

ART UNIT	PAPER NUMBER
2616	

NOTIFICATION DATE	DELIVERY MODE
05/02/2008	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

PTO@HUFFMANLAW.NET

Office Action Summary	Application No. 09/881,628	Applicant(s) MUSOLL ET AL.	
	Examiner Kevin Mew	Art Unit 2616	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _24 March 2008_.
2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.
3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-45_ is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5)☐ Claim(s) _____ is/are allowed.
6)☒ Claim(s) _1-45_ is/are rejected.
7)☐ Claim(s) _____ is/are objected to.
8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.
10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a)☐ All b)☐ Some * c)☐ None of:
1.☐ Certified copies of the priority documents have been received.
2.☐ Certified copies of the priority documents have been received in Application No. _____.
3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _2/20/2008_.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

Detailed Action

Response to Amendment

1. Applicant's Remarks/Arguments filed on 3/24/2008 regarding claims 1-39 have been considered. Claims 40-45 have been newly added by applicant. Claims 1-45 are currently pending.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

2. Claims 1-5, 7, 9-12, 13-19, 21-25, 26-36, 38-39 are rejected under 35 U.S.C. 102(e) as being anticipated by Nemirovsky et al. (US Publication 2002/0062435 A1).

Regarding claim 1, Nemirovsky discloses a context-selection mechanism within a processor (**within a multi-streaming processor**, element 14, Fig. 2) for selecting a context (**determining a context frame based on priority**, paragraphs 0041, 0061) from a pool of contexts (**from a plurality of context frames**, paragraph 0041) for processing a data packet (**for processing packet stream**, paragraph 0041) comprising:

an interface receiving a data packet (**data cache for receiving data**, paragraph 0040 and Fig. 2) and communicating with a multi-streaming processor (**for communicating with a multi-streaming processor**, paragraph 0040) said multi-streaming processor (**multi-streaming**

processor, element 14, Fig. 2) comprising a core that includes the pool of contexts (**comprising a core that includes the pool of contexts**, paragraph 0041 and Fig. 2);

circuitry (**priority control unit**, element 9, Fig. 2) for computing input data into a result value according to logic rule (**for computing a priority according to monitoring the processing resources**, paragraph 0066) and for selecting a context based on the computed value (**for selecting a stream context based on the computed priority**, paragraphs 0065, 0066); and

a loading mechanism for preloading data corresponding to the received data packets into the selected context (**loading data through the load/store units**, paragraph 0042) for subsequent processing (**for subsequent processing in accordance with instructions**, paragraph 0042);

wherein computation of the input data functions (**the computation of processing resources such as the frequency access to functional units, memory and caches**, paragraph 0066**)** to enable identification and selection of a best context for packet processing according to the logic rule at the instant time (**enables the identification and selection of a context for processing data packets according to the change in priority associated with the context**, paragraphs 0019, 0020) such that a multitude of context selections made over a period of time (**such that the context selections are made over a processor cycle**, paragraph 0066) facilitates balancing of load pressure on functional units housed within the multi-streaming processor core and required for packet processing (**dynamically facilitates load balancing on utilization of resources housed within the multi-streaming processor core required for packet processing**, paragraph 0066).

Regarding claim 2, Nemirovsky discloses the context-selection mechanism of claim 1 integrated to a data packet router operating in a data-packet-network (**integrated to a packet-data router operating in a data network**, paragraph 0028).

Regarding claim 3, Nemirovsky discloses the context-selection mechanism of claim 2 wherein the data-packet- network is the Internet network (**the Internet Protocol network**, abstract).

Regarding claim 4, Nemirovsky discloses the context-selection mechanism of claim 1 wherein the pool of contexts (**the pool of context frames**, paragraph 0041) is divided into separate clusters (**separate threads**, Fig. 1a) in the core (**multi-streaming processor core**, paragraph 0041 and Fig. 2), each cluster containing some of the functional units used in packet processing (**each thread containing instructions used in processing the streams**, paragraph 0041).

Regarding claim 5, Nemirovsky discloses the context-selection mechanism of claim 1 wherein the input data into the computation circuitry includes availability information of individual ones of the pool of contexts at the time of computation (**input data into the priority control unit includes access to available resources**, paragraph 0034).

Regarding claim 6, Nemirovsky discloses the method of claim 5 wherein the input data into the computation circuitry further includes real time information of any processing streams stalled in un-available ones of the pool of contexts (includes whether the streams are disabled, paragraph 0020) and the reason for the stall (in response to processing statistics or for the reason of processor interrupt, paragraph 0020).

Regarding claim 7, Nemirovsky discloses the method of claim 5 wherein the input data into the computation circuitry further includes statistical data about previous processing time periods required to process similar data packets (**includes on-chip processing statistics**, paragraphs 0019-0020).

Regarding claim 9, Nemirovsky discloses the context-selection mechanism of claim 1 wherein the input data is sourced from the multi-streaming processor (**input data is sourced from within multi-streaming processor**, paragraph 0066).

Regarding claim 10, Nemirovsky discloses the context-selection mechanism of claim 1 wherein the input data is sourced from a third party (**sourced from other system components or software control**, paragraph 0066).

Regarding claim 11, Nemirovsky discloses the context-selection mechanism of claim 4 wherein the clusters are numbered (**threads are numbered**, paragraph 0036 and elements 1, 2, 3, Fig. 1) and the functional units are distributed symmetrically therein (**resources are**

distributed symmetrically among active threads 1 and 2, paragraph 0036, and elements 10-13, Fig. 1).

Regarding claim 12, Nemirovsky discloses the context-selection mechanism of claim 4 wherein the clusters are numbered (**threads are numbered**, paragraph 0036 and elements 1, 2, 3, Fig. 1) and the functional units are distributed asymmetrically therein (**resources are not distributed to inactive thread 3**, paragraph 0036 and Fig. 1).

Regarding claim 13, Nemirovsky discloses a system for load balancing pressure on functional units (**dynamically facilitates load balancing on utilization of resources**, paragraph 0066) within a multi-streaming processor core (**within a multi-streaming processor core**, element 14, Fig. 2) during the processing of multiple data packets (**for processing multiple packet streams**, paragraph 0041) comprising:

a context-selection mechanism having a communication interface (**a mechanism for determining a context frame having a data cache**, paragraphs 0040, 0041, 0061);

circuitry (**priority control unit**, element 9, Fig. 2) for computing input data into a result value according to logic rule (**for computing a priority according to monitoring the processing resources**, paragraph 0066) and a mechanism for preloading data corresponding to a data packet received from the communication interface within the core (**loading data through the load/store units**, paragraph 0042) into available ones of a pool of contexts (**into a plurality of contexts associated with the data streams**, paragraph 0041);

a multi-streaming processor core (**multi-streaming processor 14**, element 14, Fig. 2) responsible for processing the data packets (**for processing packet information via the Internet**, see col. 5, lines 23-49), the processor core hosting the functional units (**the processor core hosting the functional units of integer units, floating point units, multi-threaded fetch unit, instruction scheduler, priority controller and so on**, paragraph 0042 and Fig. 2) and the context pool (**the pool of context frames**, paragraph 0041); and

a set of instructions comprising the one or more logic rules governing context selection (**instructions comprising computing a priority according to monitoring the processing resources**, paragraph 0066), wherein packet processing pressure upon the functional units within the processor core is balanced (**dynamically facilitates load balancing on utilization of resources housed within the multi-streaming processor core required for packet processing**, paragraph 0066) by selecting individual contexts for processing packet information based at least in part on the value (**by selecting the contexts for processing data packets based on priority associated with the context**, paragraphs 0019, 0020).

Regarding claim 14, Nemirovsky discloses the context-selection mechanism of claim 13 integrated to a data packet router operating in a data-packet-network (**integrated to a packet-data router operating in a data network**, paragraph 0028).

Regarding claim 15, Nemirovsky discloses the context-selection mechanism of claim 14 wherein the data-packet- network is the Internet network (**the Internet Protocol network**, abstract).

Regarding claim 16, Nemirovsky discloses the context-selection mechanism of claim 13 wherein the pool of contexts (**the pool of context frames**, paragraph 0041) is divided into separate clusters (**separate threads**, Fig. 1a) in the processing unit (**multi-streaming processor**, paragraph 0041 and Fig. 2), each cluster containing some of the functional units used in packet processing (**each thread containing instructions used in processing the streams**, paragraph 0041).

Regarding claim 17, Nemirovsky discloses the context-selection mechanism of claim 13 wherein the input data into the computation circuitry includes availability information of individual ones of the pool of contexts at the time of computation (**input data into the priority control unit includes access to available resources**, paragraph 0034).

Regarding claim 18, Nemirovsky discloses the context-selection mechanism of claim 13 wherein the input data into the computation circuitry further includes real time information of any processing streams stalled in un-available ones of the pool of contexts (**real-time information of the failure of a streaming server based on determining that the received bit rate**, see col. 10, lines 1-18) and the reason for the stall (**when the bit rate is below a threshold**, see col. 10, lines 1-18).

Regarding claim 19, Nemirovsky discloses the context-selection mechanism of claim 13 wherein the input data into the computation circuitry further includes statistical data about

previous processing time periods required to process similar data packets (**includes on-chip processing statistics**, paragraphs 0019-0020).

Regarding claim 21, Nemirovsky discloses the context-selection mechanism of claim 13 wherein the input data is sourced from the multi-streaming processor (**input data is sourced from within multi-streaming processor**, paragraph 0066) and provided in a software table (**priority table**, paragraph 0044).

Regarding claim 22, Nemirovsky discloses the context-selection mechanism of claim 13 wherein the input data is sourced from a third party (**sourced from other system components or software control**, paragraph 0066).

Regarding claim 23, Nemirovsky discloses the context-selection mechanism of claim 16 wherein the clusters are numbered (**threads are numbered**, paragraph 0036 and elements 1, 2, 3, Fig. 1) and the functional units are distributed symmetrically therein (**resources are distributed symmetrically among active threads 1 and 2**, paragraph 0036, and elements 10-13, Fig. 1).

Regarding claim 24, Nemirovsky discloses the context-selection mechanism of claim 16 wherein the clusters are numbered (**threads are numbered**, paragraph 0036 and elements 1, 2, 3, Fig. 1) and the functional units are distributed asymmetrically therein (**resources are not distributed to inactive thread 3**, paragraph 0036 and Fig. 1).

Regarding claim 25, Nemirovsky discloses the system of claim 13 wherein the set of instructions comprising the logic rule is programmable (**programmable**, paragraph 0073).

Regarding claim 26, Nemirovsky discloses a method for load balancing pressure on functional units (**dynamically facilitates load balancing on utilization of resources**, paragraph 0066) contained within a multi-streaming processor core (**within a multi-streaming processor core**, element 14, Fig. 2) during the processing of multiple data packets (**for processing multiple packet streams**, paragraph 0041) comprising steps of:

(a) arranging the functional units (**arranging the resources**, Fig. 2) into more than one separate cluster (**into multiple threads**, Fig. 1) on the core of the processor (**on the core of the multi-streaming processor**, element 14, Fig. 2), each cluster (**each thread**, Fig. 1a) containing an equal number of contexts (**contains a context frame**, Fig. 1a) that may write to the functional units (**to resources 10-13,** Fig. 1) within the hosting cluster (**within the processing core**, Fig. 2), wherein said functional units and contexts are included in the processor core (**threads and context frames are included in the processor core,** paragraph 0041 Fig. 2);

(b) receiving a data packet for processing (**data cache for receiving data streams for processing**, paragraph 0040 and Fig. 2);

(c) receiving as input for computation, data about the instant availability status of individual contexts within each cluster (**includes whether the streams are disabled**, paragraph 0020);

(d) receiving as input for computation, data about stream status of streams occupying any contexts within each cluster (**receiving information about the context frame associated with each thread**, paragraph 0041); and

(e) computing the data received as input to produce a value (**for computing a priority according to monitoring the processing resources**, paragraph 0066), the value identifying and initiating selection of a context for processing packet information from the received data packet (**enables the identification and selection of a context for processing data packets according to the change in priority associated with the context**, paragraphs 0019, 0020) and balancing packet processing load of the functional units within each cluster (**dynamically facilitates load balancing on utilization of resources**, paragraph 0066); and

(f) preloading data corresponding to the received data packets into the selected context (**loading data through the load/store units into a selected context frame**, paragraphs 0041, 0042 and element 210, Fig. 2) for subsequent processing (**for subsequent processing in accordance with instructions**, paragraph 0042);

(g) repeating steps (b) through (f) for each of the multiple data packets for processing (**the process of priority code assignment for the streams is in a repeating manner**, paragraph 0010).

Regarding claim 27, Nemirovsky discloses the method of claim 26 integrated to a data packet router operating in a data-packet-network (**integrated to a packet-data router operating in a data network**, paragraph 0028).

Regarding claim 28, Nemirovsky discloses the method of claim 27 wherein the data-packet- network is the Internet network (**the Internet Protocol network**, abstract).

Regarding claim 29, Nemirovsky discloses the method of claim 26 wherein in step (a) the functional units are provided within each cluster in a symmetrical fashion (**resources are distributed symmetrically among active threads 1 and 2**, paragraph 0036, and elements 10-13, Fig. 1).

Regarding claim 30, Nemirovsky discloses the method of claim 26 wherein in step (a) the functional units are provided within each cluster in an asymmetrical fashion (**resources are not distributed to inactive thread 3 while resources are distributed symmetrically among active threads 1 and 2**, paragraph 0036 and Fig. 1).

Regarding claim 31, Nemirovsky discloses the method of claim 26 wherein in step (b) the packet is received at a data port of a data router and requires automatic activation (**data stream is received at a port of a data router and is assigned active priority**, paragraphs 0028 and 0041).

Regarding claim 32, Nemirovsky discloses the method of claim 26 wherein in step (b) the packet is held by the processor and requires a context for processing (**requires context frame for processing**, paragraph 0041).

Regarding claim 33, Nemirovsky discloses the method of claim 26 wherein in step (c) availability status comprises an indication of which one of two components owns each context (the list of numbers for each stream stores a stream's claim to individual resources, paragraph 0056).

Regarding claim 34, Nemirovsky discloses the method of claim 33 wherein in step (c) one of the components is the processor (resource is the processor with integer and floating point units) and other component is a packet management unit (other resource is memory, paragraph 0066, and Fig. 2).

Regarding claim 35, Nemirovsky discloses the method of claim 26 wherein in step (d) the data about stream status includes whether or not streams are stalled within any of the contexts (includes whether the streams are disabled, paragraph 0020) and the reason for the stall (in response to processing statistics or for the reason of processor interrupt, paragraph 0020).

Regarding claim 36, Nemirovsky discloses the method of claim 26 wherein in step (d) the data about stream status includes time parameters of how long each stream will take to process data packets associated with their contexts (time intervals to show how long each thread will take to process data frames, paragraph 0046).

Regarding claim 38, Nemirovsky discloses the method of claim 26 wherein in steps (c) through (d) are practiced according to a set of rules of logic (according to monitoring the processing resources, paragraph 0066).

Regarding claim 39, Nemirovsky discloses the method of claim 39 wherein the rule of logic is programmable (programmable, paragraph 0073).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 8, 20, 37 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nemirovsky et al. in view of Zisapel et al. (USP 6,249,801).

Regarding claim 8, Nemirovsky discloses all the aspects of the claimed invention set forth in the rejection of claim 5 above, except fails to explicitly show the context-selection mechanism of claim 5 wherein the input data into the computation circuitry further includes statistical data about previous processing time periods required to process similar data packets.

However, Zisapel discloses the request type from client can be DNS request or HTTP request (see col. 7, lines 52-61).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the load balancing method and system of Nemirovsky with the teaching of Zisapel in rerouting requests based on the type of request made by client such that the weighted value to determine which load balancer will be the best proximity network to respond to client's request is based on the distribution of what request type to be instructed by client.

The motivation to do so is to enable redirecting the request to the appropriate location according to the type of request made by client during the determination of the best proximity network to client.

Regarding claim 20, Nemirovsky discloses all the aspects of the claimed invention set forth in the rejection of claim 13 above, except fails to explicitly show the system of claim 13 wherein the input data into the computation circuitry further includes statistical data about the distribution of instruction types associated with individual ones of previously processed and similar data packets. However, Zisapel discloses the request type from client can be DNS request or HTTP request (see col. 7, lines 52-61). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the load balancing method and system of Nemirovsky with the teaching of Zisapel in rerouting requests based on the type of request made by client such that the weighted value to determine which load balancer will be the best proximity network to respond to client's request is based on the distribution of what request type to be instructed by client. The motivation to do so is to enable redirecting the

request to the appropriate location according to the type of request made by client during the determination of the best proximity network to client.

Regarding claim 37, Nemirovsky discloses all the aspects of the claimed invention set forth in the rejection of claim 26 above, except fails to explicitly show the method of claim 26 wherein in step (d) the data about stream status includes distribution parameters of instruction types that each stream has executed to process its data packet.

However, Zisapel discloses the request type from client can be DNS request or HTTP request (see col. 7, lines 52-61).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the load balancing method and system of Nemirovsky with the teaching of Zisapel in rerouting requests based on the type of request made by client such that the weighted value to determine which load balancer will be the best proximity network to respond to client's request is based on the distribution of what request type to be instructed by client. The motivation to do so is to enable redirecting the request to the appropriate location according to the type of request made by client during the determination of the best proximity network to client.

4. Claims 40-45 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nemirovsky et al. in view of Ludtke et al. (US Publication 2003/0210252).

Regarding claim 40, Nemirovsky discloses all the aspects of claim 1 above, except fails to explicitly show the context-selection mechanism of claim 1, wherein the data corresponding to the received data packet includes:

an address indicating the start of a memory region in which the received data

packet is stored; and at least a portion of a header of the received data packet.

However, Ludtke discloses data packet that includes a header and a memory address for identifying where the data is to be stored (paragraph 0071).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the data packet of Nemirovsky with the teaching of Ludtke such that the data packets of Nemirovsky would include a header and a memory address for identifying where the data is to be stored.

The motivation to do so is to allow the header information of the data packet to be striped by a stream processor to determine the appropriate memory location to store/buffer the data received.

Regarding claim 41, Nemirovsky and Ludkte disclose all the aspects of claim 40 above. Nemirovsky does not explicitly show the context-selection mechanism of claim 40, wherein the loading mechanism is configured to:

convey a first indication to the multi-streaming processor at the start of preloading

data corresponding to the received data packet; and convey a second indication to the multi-streaming processor at the completion of preloading data corresponding to the received data packet.

However, Ludtke discloses that stripping a header in the data packet indicates the start of the process of loading data in a memory location and sending a trigger packet indicates the end of the stream of data (paragraph 0071).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the data packet of Nemirovsky with the teaching of Ludtke such that the data packets of Nemirovsky would sending a trigger packet indicates the end of the loading of the stream of data.

The motivation to do so is to allow the transfer of the data to another device for further processing.

Regarding claim 42, Nemirovsky discloses all the aspects of claim 13 above, except fails to explicitly show the context-selection mechanism of claim 13, wherein the data corresponding to the received data packet includes:

an address indicating the start of a memory region in which the received data

packet is stored; and at least a portion of a header of the received data packet.

However, Ludtke discloses data packet that includes a header and a memory address for identifying where the data is to be stored (paragraph 0071).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the data packet of Nemirovsky with the teaching of Ludtke such that the data packets of Nemirovsky would include a header and a memory address for identifying where the data is to be stored.

The motivation to do so is to allow the header information of the data packet to be striped by a stream processor to determine the appropriate memory location to store/buffer the data received.

Regarding claim 43, Nemirovsky and Ludkte disclose all the aspects of claim 42 above. Nemirovsky does not explicitly show the context-selection mechanism of claim 42, wherein the loading mechanism is configured to:

convey a first indication to the multi-streaming processor at the start of preloading

data corresponding to the received data packet; and convey a second indication to the multi-streaming processor at the completion of preloading data corresponding to the received data packet.

However, Ludtke discloses that stripping a header in the data packet indicates the start of the process of loading data in a memory location and sending a trigger packet indicates the end of the stream of data (paragraph 0071).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the data packet of Nemirovsky with the teaching of Ludtke such

that the data packets of Nemirovsky would sending a trigger packet indicates the end of the loading of the stream of data.

The motivation to do so is to allow the transfer of the data to another device for further processing.

Regarding claim 44, Nemirovsky discloses all the aspects of claim 26 above, except fails to explicitly show the context-selection mechanism of claim 26, wherein the data corresponding to the received data packet includes:

an address indicating the start of a memory region in which the received data

packet is stored; and at least a portion of a header of the received data packet.

However, Ludtke discloses data packet that includes a header and a memory address for identifying where the data is to be stored (paragraph 0071).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the data packet of Nemirovsky with the teaching of Ludtke such that the data packets of Nemirovsky would include a header and a memory address for identifying where the data is to be stored.

The motivation to do so is to allow the header information of the data packet to be striped by a stream processor to determine the appropriate memory location to store/buffer the data received.

Regarding claim 44, Nemirovsky and Ludkte disclose all the aspects of claim 44 above. Nemirovsky does not explicitly show the context-selection mechanism of claim 44, wherein the loading mechanism is configured to:

convey a first indication to the multi-streaming processor at the start of preloading data corresponding to the received data packet; and convey a second indication to the multi-streaming processor at the completion of preloading data corresponding to the received data packet.

However, Ludtke discloses that stripping a header in the data packet indicates the start of the process of loading data in a memory location and sending a trigger packet indicates the end of the stream of data (paragraph 0071).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the data packet of Nemirovsky with the teaching of Ludtke such that the data packets of Nemirovsky would sending a trigger packet indicates the end of the loading of the stream of data.

The motivation to do so is to allow the transfer of the data to another device for further processing.

Response to Arguments

5. Applicant's arguments filed on 3/24/2008 with respect to claims 1-39 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kevin Mew whose telephone number is 571-272-3141. The examiner can normally be reached on 9:00 am - 5:30 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Chi Pham can be reached on 571-272-3179. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Chi H Pham/
Supervisory Patent Examiner, Art Unit 2616
4/28/08

Kevin Mew /K. M./
Work Group 2616